Title:	Document Version:
Deliverable D5.1 **Application Layer Requirements Derived from the Use of PLC Technology**	2.0

Project Number:	Project Acronym:	Project Title:
IST-2001-37613	6POWER	IPv6, QoS & Power Line Integration

Contractual Delivery Date:	Actual Delivery Date:	Deliverable Type* - Security**:
31/01/2003	23/06/2003	R – PU

* Type: P - Prototype, R - Report, D - Demonstrator, O - Other

** Security Class: PU- Public, PP – Restricted to other programme participants (including the Commission), RE – Restricted to a group defined by the consortium (including the Commission), CO – Confidential, only for members of the consortium (including the Commission)

Responsible and Editor/Author:	Organization:	Contributing WP:
Marco Stahlmann	MCL	WP5

Authors (organizations):

Emilio García (ASSA), Pedro M. Ruiz (ASSA), Juan Antonio Sánchez (ASSA), Jordi Palet (Consulintel), Alvaro Vives (Consulintel), Daniel Martínez (UMU), Antonio Gómez-Skarmeta (UMU).

Abstract:

This document introduces the requirements at the application layer, which become necessary by the integration of IPv6 and PLC.

The merging of PLC and IPv6 makes it necessary to adapt the application layer so that QoS mechanisms of the two worlds can work efficiently together and complement each other. Therefore the sustainable parameters achievable by the PLC network have to be gathered and adjusted to requirements of the newly developed advanced applications.

Keywords:

Adaptive Applications, IPv6, PLC, QoS.

Revision History

The following table describes the main changes done in the document since its creation.

Revision	Date	Description	Author (Organization)
v1.0	12/06/2003	Document creation	Marco Stahlmann (MCL)
v1.1	20/06/2003	Integration of contributions from ASSA and UMU into this final document	Emilio García (ASSA) Juan Antonio Sánchez (ASSA) Pedro M. Ruiz (ASSA) Daniel Martínez (UMU) Antonio G. Skarmeta (UMU)
v1.2	20/06/2003	Harmonization and extension	Marco Stahlmann (MCL)
v1.3	20/06/2003	Inserting the bandwidth broker figure	Marco Stahlmann (MCL)
v1.9	23/06/2003	Transferring the document to the actual template	Alvaro Vives (Consulintel)
v2.0	23/06/2003	Final Review	Jordi Palet (Consulintel)

Executive Summary

The document introduces the requirements at the application layer, which become necessary by the integration of IPv6 and PLC.

The merging of PLC and IPv6 makes it necessary to adapt the application layer so that QoS mechanisms of the two worlds can work efficiently together and complement each other. Therefore the sustainable parameters achievable by the PLC network have to be gathered and adjusted to requirements of the newly developed advanced applications.

The document describes the applications and their requirements within a PLC network and relates them to the PLC channel behavior. It also shows the current QoS implementation and states which QoS classes will be supported and how the bandwidth broker will manage the system.

Table of Contents

Table of Figures

1. APPLICATIONS FOR PLC NETWORKS

Several services and applications are expected to be used on top of future PLC networks: VoIP calls, network games, e-mail transfers, file downloads, videoconferences, collaborative applications, high-definition video streaming, etc. All these services have different bandwidth, latency and packet-loss-rate requirements, but all of them have to share the same network. Additionally few of these applications are able to probe for network conditions; relevant QoS variables include packet losses, delay variation and available bandwidth. Furthermore, it is clear that gathering this information is not enough: some reaction is needed in form of adaptation to the environment, if an acceptable user-perceived QoS is to be offered.

Since our goal is to deploy IPv6 over PLC, QoS is a fundamental issue that must be dealt with. Some mechanism is needed to assure the quality of service, while using as a transport a network, which does not guarantee any quality. This demonstrates the need for adaptive applications and the appropriate signaling mechanisms.

1.1 Requirements of Real-Time Applications

Real-time applications like voice/video calls or network games have very strict requirements for their transmission parameters to guarantee not perceivable quality digressions. These kinds of applications are very sensitive to variations in delay and jitter and require normally guaranteed bandwidths but are relatively robust to packet loss. To a certain degree such applications can cope with degradation in transmission quality but at a certain level the user will become aware of the quality loss. At this point the application has to be stopped or the QoS of the affected application has to be renegotiated. The former will be very unsatisfactory for the customer, as the disruption of the service will leave a high impression on the user. The latter will also not be very satisfactory for the customer but he will cope with a short-term reduction of the service when the service can be maintained, which leads to adjustable applications and renegotiable transmission parameters.

Applications involving two or more interacting users are very sensible to latency and jitter. This is due to the impossibility of the application to buffer the data stream to a large degree since the user is well aware of e.g. in voice communications the resulting echoes in the communication or stumble in the video stream. On the other hand these applications are somewhat insensible to the packet loss rate since the voice/video coder can normally cope well with single lost packets. The bandwidth is not so important for these applications since there is large variety of codec's supporting all kind of different bandwidths from low to high rates.

A special kind of interactive application are the more and more popular interactive network games where the user is very cautious of delay since he will experience severe disadvantages against his opponents in the game when his actions are not transmitted instantaneously and depending on the actual game he is also aware of lost packets since then his movements become erratic.

Other kinds of real-time applications like web browsing and video streaming are less restrictive to their requirements. Even though web browsing is interactive the user tolerates a certain delay for the start up of the connection and basically all other parameters are more or less irrelevant and only extending the download time. As for video streaming the major requirement is bandwidth since the user wants to have high quality videos, which comprise of a huge amount of data. The delay and jitter are less relevant since the user will tolerate an initial start up delay so that the application can buffer a part of the stream and compensate the jitter and bandwidth variations during the playback.

Relevant parameters for the introduced applications are listed in the following table.

Application	Bandwidth	Latency	Jitter sensitivity	Packet loss rate sensitivity
VoIP	Low 8-64kbit/s	High<30ms	High	Low-medium
Videoconference	Medium 64K-1Mbit/s	High<30ms	High	Low-medium
Network Games	64-256kbit/s	High<5ms	High	Med-high
Web browsing	64-256kbit/s	Low<2s	Low	Medium-high
Collaborative work	Medium-high 512k-10Mbit/s	High<30ms	Low	Medium-high
Video streaming	Medium-high512k-10Mbit/s	Very low<30s	Low-medium	Medium-high

Figure 1-1: Real-Time Applications Requirements

1.2 Requirements of Non-Real-Time Applications

Non-real-time applications are much more adaptable in their transmission requirements than real-time applications. For them to function normally only a minimum bandwidth is necessary. Although they have no real requirements for packet loss they are very sensitive to it in the way that lost packets have to be retransmitted since the integrity of the data is the most important. Those applications normally run in the background or over night. The main difference between those applications is the amount of data to transfer, which can vary from few bytes as by e-mail to many MBytes/GBytes for file transfer.

Application	Bandwidth	Latency	Jitter sensitivity	Packet loss rate sensitivity
E-Mail	Medium 64-256kbit/s	Low<2s	Very low	Medium-high
File Transfer	Medium-high 64k-10Mbit/s	Very low<10s	Very low	Very high

Figure 1-2: Non Real-Time Applications Requirements

1.3 The Need for Adaptive Applications

Most of the currently used applications for internetworking multimedia in IP networks are based on a similar architecture in which over the IP network layer there is a basic transport provided by UDP. Then the Real Time Protocol (RTP) is the preferred framing protocol for multimedia traffic while the Real Time Control Protocol (RTCP) is used to convey additional information about the participants, receiver information statistics, etc. Finally, there is a special middleware called codec's, which is used for compressing the raw multimedia information into a reduced format, which may be de-codified at the other end.

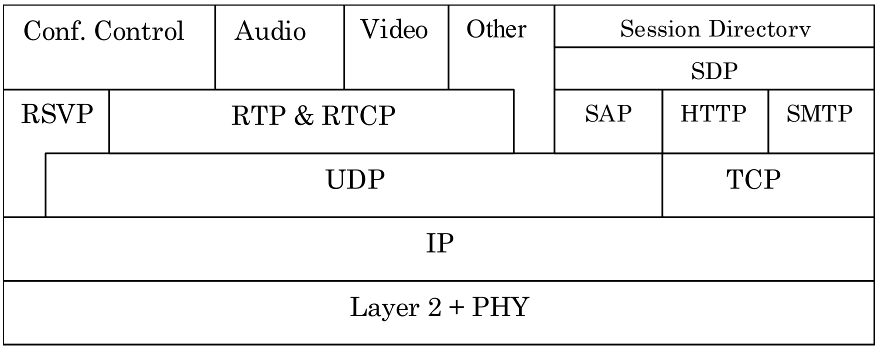

Figure 1-3: The Architecture of a Multimedia Application

This reference architecture, which is presented in Figure 1-3 shows the relation between the different components as well as their interactions. As it is shown, there are two different planes. In the control plane there are elements like the session management and definition protocols, which allow for the definition of the parameters that a conference is going to use. Those parameters are specified in terms of the codec's, which are going to be used, the sampling rates, the participants, the addresses that can be used to participate in the session, etc. This kind of control information can be sent using different transport protocols like HTTP, SMTP, etc. On the other hand, there is a data plane as well which is in charge of the transmission of the encoded multimedia content over the IP network. As previously stated, these encoded content will be encapsulated in RTP packets, which are carried using UDP.

Once the session is defined, all the participants will select the attributes defined for the session when sending and receiving multimedia data. Usually, these attributes are not changed during the life of the session. This means that if the network gets congested it is difficult for the application to offer a good quality. If the bandwidth becomes so scarce that the quality obtained is unacceptable, the participants may eventually renegotiate the session, but this would take some time, it is not straightforward for the users and it is not seamless enough as to be used in future mobile networks in which the user is not expecting their communications to be suddenly interrupted.

The example of plugging a new device is just one of the multiple cases in which the performance and the quality might be strongly reduced. There are many other circumstances, which might cause these applications to give a bad quality to the user. In these networks in which the network there is a lot of interference and there are not any mechanisms facilitating the reservation any network layer QoS, the standard applications are not expected to offer a good service.

The most relevant parameters, which affect the operation of these applications, are as follows:

- Limited and highly variable bandwidth.
- Burst packet losses due to link limited instantaneous bandwidth.
- Variable end-to-end delay.
- Interferences in at the link layer.

The important question is, what can a traditional IP multimedia application do in order to improve this QoS? As we have seen in the problems above, most of them cannot be resolved by the provision of network layer QoS. So, the only approach to improve the QoS would be to make it at the application layer. The only mechanism that a traditional IP multimedia application has for getting any clue on the quality information coming from the other end is the information provided by RTCP. However, RTCP was thought for low bandwidth Internet networks so it is designed so that not too much bandwidth is consumed. This means that usually an RTCP report is sent approximately every 5 seconds but this rate may be lower during the session. These rates are not appropriate for these highly changing scenarios. In addition, most of the existing applications are able to generate and receive RTCP reports but very few of them change its behavior when reports indicate a bad behavior.

Our proposal for such variable scenarios is to look at applications, which are able to adapt to the network conditions to offer a better user-perceived QoS. That is, provided that the network conditions cannot be improved, if the application is provided with enough information about the end to end quality which is being perceived it can dynamically adapt its capabilities like audio and video codec's, frame sizes, frame rates, etc. to make the user get a minimum level of QoS. The key heuristic is that in a degraded network condition, the user may prefer having a smaller video size and a good quality audio rather than a slow motion high-resolution video with an interrupted and unintelligible audio stream.

Examples of such adaptations might include among others the following dynamic changes to the following settings:

- Codec's used for audio and video. This allows the application to decrease the bandwidth consumption. Usually less bandwidth means less quality. However in these environments the user prefers less quality audio or video than packet losses, which may cause the audio and video tools not to be useful.
- Audio sampling rate. The sampling rate is proportional to the number of packets, which will be sent out to the network. Higher sampling rates mean better quality and higher bandwidth consumption.
- Video size. The bigger the video size, the higher the bandwidth consumption. In scarce bandwidth environments the user will prefer seen smaller videos than bad quality ones in which most of the frames are lost.
- Frames per second. Transmitting at a lower frame rate means saving bandwidth. In most situations the application does not need to reach the optimum number of 25 fps to make the user 'feel' a fair quality. In addition, practical limitations from video cameras make the subjective better sensation be achieved by using 24 fps dividers: 1, 2, 3, 4, 6, 8, 12 and 24 fps. This means for example that the user perceives better quality when switching from 12 to 8 fps instead of changing from 12 to 10 and changing the codec as well.
- Buffering. Intelligent and dynamically adaptable buffers may help offering a better quality in adverse network conditions.
- Components to use. In very constrained bandwidth scenarios in which even using the lowest bandwidth consumption approaches, the user may prefer using some components instead of using all of them with a poor quality. For example, in a 40 Kbps scenario the user may prefer just receiving a GSM audio flow without any losses rather than receiving a video with very poor quality and an unintelligible audio stream.

These kinds of applications might turn out to be a key piece in future PLC networks. So, as future work, our intention is to analyze its use in the framework of this project, and over the PLC test-beds set up within WP3.

2. PLC CHANNEL BEHAVIOR

The power line was originally designed for distribution of 220/110 volts power at 50-60 Hz. Using this medium for broadband communications at higher frequency bands presents many technical problems.

The low-voltage power line network is made of a variety of wiring types, connected in almost random ways (which has a strong effect on impedance mismatch). In addition to that, very different types of devices are part of the LV network (electricity meters, fuses, etc.) and a large variety of appliances can be connected in any point (air conditioners, washing machines, TV sets, etc.).

Those harsh conditions are common in certain types of networks, such as PLC, where the physical layer offers highly variable characteristics. The fact that PLC utilizes electric carriers to transport data makes it rather sensitive to electric noise. PLC tries to use all the carriers that are not too noisy for information transmission, making each of them transport a small flow of data. As the number of carriers, which are clear enough descends, less information can be transmitted at the same time, and thus the bandwidth decreases. This means that the available bandwidth depends on how noisy the electric signal is. Moreover, although PLC tries not to use the carriers, which are too noisy, noise can appear on a carrier that is being already used. This produces packet losses, a problem that also gets worse the noisier the power line is.

For example, Figure 2-1 shows the real case of a production PLC network, plotting the bandwidth during a whole day. While a number of applications are not significantly affected (differed services – FTP, HTTP – or even real-time services such as streaming) others such as emerging IP based videoconference applications present a need for certain guarantees: studies have shown that user-perceived audio quality starts becoming extremely bad when packet loss rate goes over 20% (even when packet retransmission techniques are applied).

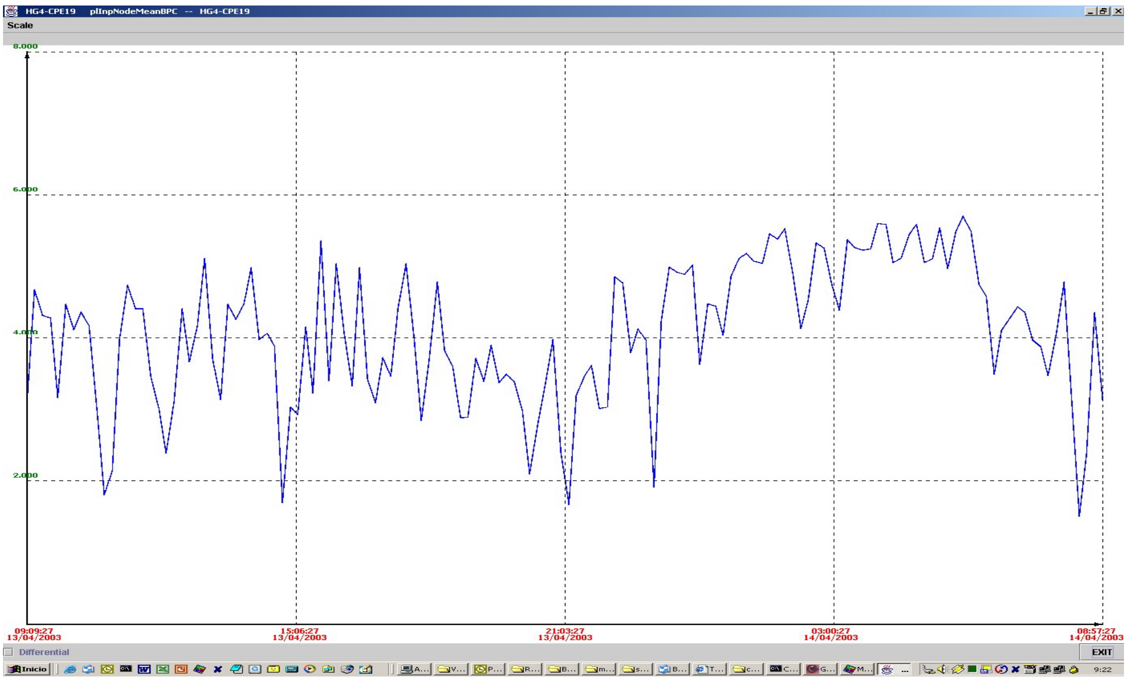

Figure 2-1: Link Variability in PLC Networks

Because of this variability it is not easy to guarantee a data transmission throughput in Mbits/second and a fixed maximum latency to a set of CPEs, and at the same time try to support the maximum number of users. It would be needed to reserve some bandwidth because some user SNR might decrease. This reserved BW cannot be used by other CPEs, so this is not a very efficient approach.

The variability of the line data rate yields a range of actual data transmission rate. The minimum value is the data rate reachable when the quality of the line is the worst one, and the maximum value when the quality is the best one. Latencies are assured independently of the line quality provided the data source doesn't exceed the maximum reachable data rate.

Because of this actual data rate variability, the following parameters are guaranteed:

- Bandwidth (in percentage): CPEs can be assigned a minimum bandwidth. This percentage is actually a portion of time and a portion of the spectrum frequency, which is assigned to the CPE. When some CPEs don't use their reserved bandwidth all other CPEs can take profit of it. This percentage will actually yield a variable bandwidth depending on the quality of the power line seen by the CPE in both paths (Upstream and Downstream). If more bandwidth is desired for a CPE, the network operator has to increase the CPE's physical BW percentage. Bandwidth percentage can be individually set for the Upstream and Downstream.
- Maximum latency (in milliseconds): CPEs can be assigned a maximum latency in the Upstream. In the Downstream, the maximum latency depends on the number of active users and at this moment it is limited to 50 milliseconds approx. Packets are guaranteed to see a latency lower than the maximum set for the CPE provided the data packets find the driver queues empty. Latency is assured if the data flow throughput doesn't exceed the QoS contract. This throughput is the actual data rate achievable with the present line quality and bandwidth percentage. We must also distinguish between access latency and active latency. Access latency is the latency seen when the CPE's state is IDLE. This is greater than the usual latency (active latency) which is the latency seen by ACTIVE CPEs. Access latency can also be limited with the CPE_ACCESS_TIME but it is a trade-off between initial access time and overall bandwidth efficiency (more polls are inserted).

3. BANDWIDTH MANAGEMENT FOR SENSITIVE APPLICATIONS

3.1 QoS Classes

So far the QoS MAC supports the following traffic types:

- VoIP: CPEs that need a low-latency CBR data flow typical of VoIP applications. A maximum latency of 144 milliseconds in the Upstream and 50 milliseconds in the Downstream is assured for this type of CPEs with a maximum throughput range of 64 to 256 Kbit/second, depending on the line quality seen by the CPE. The mean latency is typically one half of the maximum latency.
- DATA: CPEs with configurable BW percentage and latency (latency is only configurable in the Upstream, though its maximum value is known in the Downstream, 50 ms.). The BW can range from 5% to 100% in the Upstream. The maximum latency can be 144, 288 or 576 milliseconds in the Upstream and 50 ms. maximum in the Downstream. The network operator can choose any latency in the Upstream but the MAC will assign one of these values (the nearest one). In the Downstream there are no restrictions in the percentage bandwidth that can be set for a CPE.
- UBR (Unspecified Bit Rate): In the Upstream, BW and latency is not configurable but a minimum QoS is assured for them, which is 5% BW and 576 ms. maximum latency. In the Downstream, all UBR CPEs share a percentage of the total bandwidth, which is reserved for this type of CPE. In case UBR CPEs don't use it, QoS CPEs will use it.
- ABR (Available Bit Rate): ABR CPEs don't have any QoS guarantees, so they will only transmit when none QoS CPE is Active. This means that if always there is at least one QoS CPE active, ABR CPEs will never transmit.

This classification will probably have to be extended to a finer level of resolution so that a larger variety of applications with their specific necessities can be efficiently supported. Since the PLC driver provided by DS2 is very flexible it will be possible to add or remove new quality classes if the necessity should arise. These new classes could be realized by additional queues inside a device that have a certain priority or to modify the access strategy to existing queues within the device or with a bandwidth broker which changes the bandwidth allocation throughout the system to accommodate the needs of each CPE.

3.2 Bandwidth Broker

The bandwidth broker is a centralized management instance inside a PLC system, which has the complete overview about the network topology, the actual reserved bandwidth on each link inside its domain and should ideally have also a real-time view of the available bandwidth on each link.

The bandwidth broker acts as an intermediate unit for call setups in a way that a call with ensured quality has to send a signaling packet to the broker which then manages each node taking part in this connection and reserves the bandwidth percentage and delay constraints necessary for the requested quality. At the moment only three different levels of quality are supported between an application can choose.

The signaling between the CPEs and the bandwidth broker is at the moment very rudimental and consists of simple UDP packets containing a string with the desired quality. This simple signaling will be replaced by a standardized signaling protocol, which will most likely be SIP. Thus enabling a larger variety of applications to make use of the quality of service support without changing the application.

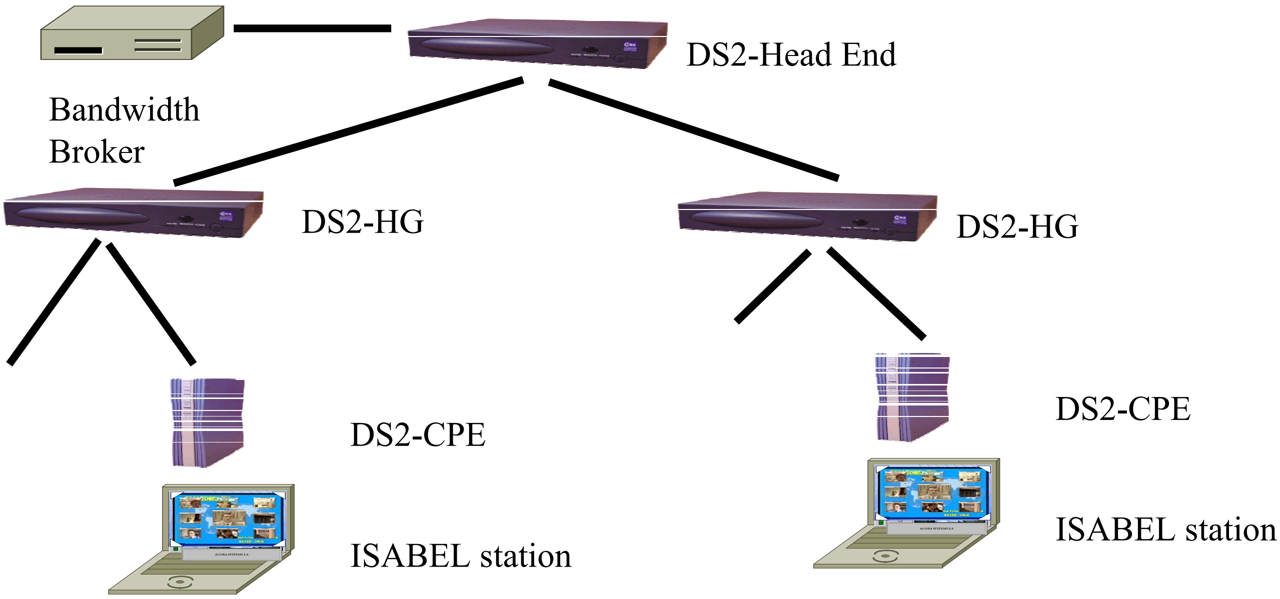

Figure 3-1: Bandwidth Broker in a PLC Network

4. SUMMARY AND CONCLUSIONS

At the current state of investigation of the application requirements in a PLC environment the results are quite promising.

The preliminary support of QoS by the current PLC driver is so far able to provide the necessary quality to handle concurrent VoIP and video connections in an environment with other competitive applications like FTP or HTTP. By further extending this functionality the bandwidth broker will be able to give QoS guarantees for various applications and take management control over the whole PLC network.

Through the introduction of adaptive applications the bandwidth broker will become even more powerful and the network can support more concurrent users since then it will be possible to renegotiate communication parameters so that all users will get their requested QoS or at least will be able to continue their service instead of a deterioration of their perceived quality.

5. REFERENCES

[1] J.C. Bolot and A. Vega-Garcia. "The case for FEC-Based Error Control for Packet Audio in the Internet". ACM Multimedia Systems, 1998.